Cyber Security Training Overview

Contents

Cyber Security Training Courses

NCC Group understands that education is an essential part of any product deployment and have developed a set of training courses designed to assist you. These courses are run by qualified security professionals with real world hands-on experience of the technologies.

NCC Group offers a broad range of training courses to address your cyber security education concerns:

- How to secure and assess the three most prolific cloud providers and the ever-present containers.
- A series of courses with a focus on assessing and securing Web, Android and iOS applications.
- Security in Software Development Lifecycle for those more concerned about establishing a governance framework and appropriate processes and policies.
- Finally, a series of courses with more of a research focus.

If you have a need to upskill your staff to either develop secure applications or identify security flaws, NCC Group has a course for you.

About NCC Group

NCC Group is a global expert in cyber security and risk mitigation, working with businesses to protect their brand, value and reputation against the ever-evolving threat landscape.

With our knowledge, experience and global footprint, we are best placed to help businesses identify, assess, mitigate & respond to the risks they face.

AWS Security Review

Ever more enterprises are moving their operations to the cloud, with AWS being the largest player in the market at present. But how can you be sure the cloud environment is safe against hackers?

The information provided in this course will be useful for both penetration testers and infrastructure engineers of an AWS account, covering interesting attack vectors and providing details on how a healthy AWS environment should be configured to prevent, mitigate and identify attacks if and when they occur.

The topics discussed in this course are beneficial for those aiming to achieve the official Security Specialty certification available from AWS.

Who should take this course?

NCC Group's AWS security review training is a focused course for security consultants and cloud architects interested in learning how to assess the main elements of a cloud environment based in AWS. We will cover the techniques and tools necessary to perform a thorough security review and provide an understanding of the major risks, along with the best security practices that should be considered when designing a cloud infrastructure.

Requirements

Participants are expected to have some familiarity with Linux, the Linux command line and basic IP networking knowledge.

Students should bring a laptop with an SSH client installed. The laptop should not have any corporate security software which restricts Internet access or forces use of a corporate proxy server for browsing.

Recommended setups are: Python installed on Linux, MacOS, Windows with WSL or MobaXTerm (PuTTY is possible, but requires extra setup).

Deliverables

PDF presentation of over 300 slides covering all modules of the training course.

Agenda - 3 days

Introduction to AWS basics

Hands on with the AWS command line and tools like Scout2 and prowler

AWS networking and common issues

Identity and Access Management (IAM) in-depth including sample policies and interesting attack vectors

The EC2 service and finding credentials with simple commands

Protecting files stored on S3 and common attacker vectors

Relational Database Service (RDS)

Monitoring and logging

Scripting with AWS

A typical assessment against an AWS environment + CTF

GCP Security Review

Ever more enterprises are moving their operations to the cloud, with customer adoption of Google Cloud Platform (GCP) steadily increasing. How can you ensure your cloud environment is secure?

NCC Group's GCP security review training is a three-day course dedicated to security consultants and cloud architects interested in learning the principal elements of an environment based in Google's cloud. It will discuss the techniques and tools necessary to perform a thorough security review and provide an understanding of the major risks, along with security best practices.

Who should take this course?

NCC Group's Google Cloud Platform (GCP) security review training is a focused course for security consultants and cloud architects interested in learning how to assess the main elements of a cloud environment based in GCP. We will cover the techniques and tools necessary to perform a thorough security review and provide an understanding of the major risks, along with the best security practices that should be considered when designing a cloud infrastructure.

Requirements

Participants are expected to have some familiarity with Linux, the Linux command line and basic IP networking knowledge. Familiarity with GCP or a similar platform (AWS, Azure, etc.) is beneficial but not necessary.

Students should bring a laptop with an SSH client installed. The laptop should not have any corporate security software which restricts Internet access or forces use of a corporate proxy server for browsing.

Recommended setups are: Python installed on Linux, MacOS, Windows with WSL or MobaXTerm (Putty is possible but requires extra setup).

Deliverables

PDF presentation of over 300 slides covering all modules of the training course.

Agenda - 3 days

The course is a mixture of presentations and hands-on exercises and aims to ensure that participants gain practical experience assessing and securing Google Cloud Platform environments.

The course includes:

- An introduction to GCP for people new to the platform, including general concepts and a comparison with other cloud providers
- How to interact with GCP through the Cloud Console, CLI tool and SDK
- An extensive discussion on the Identity and Access Management services with samples of policies and interesting attacks vectors
- A review of networking in GCP, including typical topologies and common issues
- A detailed look at the core services for computation, storage, databases, security and logging & monitoring
- Tools which can help assess and secure GCP deployments

Azure Security Review

Ever more enterprises are moving their operations to the cloud, with customer adoption of Azure steadily increasing. How can you ensure your cloud environment is secure?

NCC Group's Azure security review training is a three-day course dedicated to security consultants and cloud architects interested in learning the principal elements of an environment based in Azure's cloud. It will discuss the techniques and tools necessary to perform a thorough security review and provide an understanding of the major risks, along with security best practices.

The topics discussed in this course are beneficial for those aiming to achieve the official certification paths available from Azure.

Who should take this course?

NCC Group's Azure security review training is a focused course for security consultants and cloud architects interested in learning how to assess the main elements of a cloud environment based in Azure. We will cover the techniques and tools necessary to perform a thorough security review and provide an understanding of the major risks, along with the best security practices that should be considered when designing a cloud infrastructure.

Requirements

Participants are expected to have some familiarity with the PowerShell command line and basic IP networking knowledge.

Students should bring a laptop with an SSH client installed. The laptop should not have any corporate security software which restricts Internet access or forces use of a corporate proxy server for browsing.

Recommended setups are: PowerShell installed on Linux, MacOS or Windows.

Deliverables

PDF presentation of over 250 slides covering all modules of the training course.

Agenda - 3 days

The course is a mixture of presentations and hands-on exercises and aims to ensure that participants gain practical experience assessing and securing Azure environments.

The course includes:

- An introduction to Azure for people new to the platform
- How to interact with Azure through the Azure Portal, Azure Security Centre, Azure Cloud Shell and Azure PowerShell CMDLets
- An extensive discussion on the Identity and Access Management services with samples of policies and interesting attacks vectors
- A review of networking in Azure, including typical topologies and common issues
- A detailed look at the core services for computation, storage, databases, security and logging & monitoring
- Tools which can help assess and secure Azure deployments

Offensive Cloud Security

While security awareness and collective experience regarding the Cloud has been steadily improving, one common difficulty is applying theoretical knowledge to real-life scenarios. This training's goal is to help attendees bridge this gap by understanding how conventional technologies integrate with Cloud solutions. The training is scenario-based and focusses on applied exercises.

Attendees will experience first-hand how security vectors that exist in such eco-systems present opportunities for abuse. Throughout the training, we will also cover detection and mitigation of the attacks covered in the course.

Who should take this course?

The training is tailored towards individuals who have some experience with "the Cloud", seeking to improve their proficiency at assessing and improving the security of cloud hosted applications and infrastructures.

The training is structured as a sequence of scenarios, which mix theory and practical exercises. The theory is imparted gradually, and attendees are be given time to think for themselves and work through the exercises.

Requirements

Attendees should have at least minimal experience with a Cloud provider and be familiar with the steps required to assess and improve the security of applications and infrastructures (not necessarily cloud hosted).

Students should bring a laptop with an SSH and RDP client installed. The laptop should not have any corporate security software which restricts Internet access or forces use of a corporate proxy server for browsing.

Attendees will be provided access to virtual instances with all the required tooling.

Deliverables

PDF presentation of over 400 slides covering all modules of the training course.

Agenda - 2 days

Introduction to the Multi-Cloud

Overview of security in the Cloud

Scenarios (non-exhaustive):

- Enumerating publicly accessible resources
- Leveraging CI/CD systems to gain a foothold into Cloud environments
- Lateral movement and privilege escalation
- Compromising Azure Applications
- Pivoting around Hybrid Clouds
- Compromising cloud-synched Active Directory deployments
- A Blue Team Perspective
- Tying it all together (CTF)

Mastering Container Security

Containers and container orchestration platforms such as Kubernetes are on the rise throughout the IT world, but how do they really work and how can you attack or secure them?

This course takes a deep dive into the world of Linux containers, covering fundamental technologies and practical approaches to attacking and defending container-based systems such as Docker and Kubernetes.

In the 2020 version of the course we'll be focusing more on Kubernetes as it emerges as the dominant core of cloud native systems and looking at the wider ecosystem of products which are used in conjunction with Kubernetes.

Who should take this course?

This course is suitable for either offensive or defensive security professionals and is designed to get you up to speed with hands-on experience of how to setup, defend and attack containerized environments.

The course is quite hands-on so likely best suits attendees who will be directly using or attacking container-based systems.

Requirements

Participants are expected to have some familiarity with Linux, the Linux command line, basic IP networking and Docker knowledge.

Students should bring a laptop with an SSH client installed. The laptop should not have any corporate security software which restricts Internet access or forces use of a corporate proxy server for browsing.

Recommended setups are Linux, MacOS, Windows with WSL or MobaXTerm (Putty is possible but requires extra setup).

Deliverables

PDF presentation of over 400 slides split into the eight modules covered during the training.

Additional bonus modules covering Docker & Kubernetes ecosystem security, Windows containers and security tooling.

Agenda - 3 days

Docker Basics

Creating Docker Images

Container Fundamentals

Docker Security

Introduction to Kubernetes

Kubernetes Networking

Kubernetes Basic Security

Kubernetes Authentication & Authorisation

Kubernetes Policy Security

Kubernetes Ecosystem

Extras & CFF

Hacking & Securing Web Applications

Web applications are on the rise and the technologies they are based upon constantly evolving, but how do they really work and how can you attack or secure them?

The Hacking and Securing Web Applications course covers the methodology to assess the security of a web application and provides detailed guidance on secure development, relating to both the design and implementation of web applications.

The course is a mix of presentations and hands-on labs sessions where you can practice and experience how application vulnerabilities are detected and exploited by attackers and how applications can successfully defend against these attacks.

Who should take this course?

Hacking and Securing Web Applications is a course aimed at software developers, software architects, security consultants and quality assurance engineers who want to understand how attackers uncover and exploit vulnerabilities in web applications, and what can be done by developers to prevent it.

Requirements

Participants are expected to have some familiarity with basic web technologies like HTTP and JavaScript.

Hands-on experience in application design or development is preferred.

Participants are to bring own laptop running Windows 7 or above with local administration rights and VMware Player pre-installed.

Code examples in the course are written in Java, .NET, Java Script and PHP however the majority of the training content is platform and language agnostic and is relevant to every web application so no prior knowledge of any of these languages is required.

Deliverables

PDF presentation of over 400 slides split into over ten modules covered during the training.

Virtual machine with all hands-on practical labs discussed during the training and more.

Agenda - 3 days

Breaking and building robust authentication and authorisation mechanisms and session management

Uncovering and exploiting SQL injection, filter bypasses, query chaining and blind exploitation

Interacting securely with database management systems (DBMS)

JavaScript and thick client components reverse engineering and client-side controls bypass

Detecting and exploiting cross-site scripting to log keystrokes, port scan the victim's computer and network and execute custom payloads

Avoiding cross-site scripting, validating user input and other client-side flaws

Uncovering business logic flaws with dynamic and static code analysis

Android Application Assessment

Native mobile applications have become an expected part of any online service offering in today's connected world. While often sharing technology and functionality with web applications, they present a distinct and unique set of security concerns that need to be specifically addressed.

The course covers the fundamentals of Android's architecture and security model, the features provided to developers to help them secure corporate and personal information, and the additional measures that applications can take to provide additional security for their users. It provides the tools and techniques required determine which protections are appropriate for a given application, and to validate that these protections are in place and effective.

The course is a mixture of presentation and hands-on exercises where you will learn the methods required to identify potential vulnerabilities in Android applications and assess their severity in context. Additional material is also available on how to identify each type of vulnerability in application code, and how it can be avoided or mitigated.

Who should take this course?

Android Application Assessment is a course aimed at penetration testers, QA engineers, and application developers who want to understand the unique security concerns and defences that apply to mobile applications in general and Android in particular.

Requirements

Participants are expected to be familiar with basic networking and Internet technologies such as HTTP. Some knowledge of web application security, and the tools used to assess it, is an advantage but not required.

No specific knowledge of application development or the Android platform is necessary.

Participants are to bring their own laptops running Windows 7 or above with local administrative access, or Linux with root privileges. Either VMware Workstation or VirtualBox should be pre-installed.

Test mobile devices will be provided for on premise courses, in the case of remote delivery participants will require their own test device to follow along with the instructor demonstrations.

Deliverables

PDF presentation of all slides covered during the training session.

Example application packages demonstrating common vulnerabilities to support hands-on exercises during the course.

Virtual machine with common open source assessment tools installed and configured.

Agenda - 3 days

Android Assessment Methodology

Android Architecture

Android Security Model

Basic interaction with an Android Device

Static Analysis of Source Code

Reverse Engineering of APK

Instrumentation and Hooking

Anti-Root Protections

Traffic Analysis

Data Storage Analysis

Cryptography Analysis

Local Authentication in Android

Log Analysis

Messaging Objects – Intents

Android Permissions

Android Activities

Deep Links and App Links

Android Services

Broadcast Receivers

Content Providers

iOS Application Assessment

Native mobile applications have become an expected part of any online service offering in today's connected world. While often sharing technology and functionality with web applications, they present a distinct and unique set of security concerns that need to be specifically addressed.

The course covers the fundamentals of the iOS security model, the features provided to developers to help them secure corporate and personal information, and the additional measures that applications can take to provide additional security for their users. It provides the tools and techniques required determine which protections are appropriate for a given application, and to validate that these protections are in place and effective.

The course is a mixture of presentation and hands-on exercises where you will learn the methods required to identify potential vulnerabilities in iOS applications and assess their severity in context. Additional material is also available on how to identify each type of vulnerability in application code, and how it can be avoided or mitigated.

Who should take this course?

iOS Application Assessment is a course aimed at penetration testers, QA engineers and application developers who want to understand the unique security concerns and defences that apply to mobile applications in general and iOS in particular.

Requirements

Participants are expected to be familiar with basic networking and Internet technologies such as HTTP. Some knowledge of web application security, and the tools used to assess it, is an advantage but not required.

No specific knowledge of application development or the iOS platform is required, except for the optional sections on implementing defences.

Participants are to bring their own laptops running Windows 7 or above with local administrative access, or Linux with root privileges. This laptop should not have any restrictions on network connectivity which cannot be disabled by the participant.

Code examples in the course are primarily written in Objective-C, with JavaScript being used in some areas for dynamic instrumentation. However, no knowledge of any development language is required for the majority of the course material.

Test mobile devices will be provided for on premise courses, in the case of remote delivery participants will require their own test device to follow along with the instructor demonstrations.

Deliverables

PDF presentation of all slides covered during the training session.

Example application packages demonstrating common vulnerabilities to support hands-on exercises during the course.

Virtual machine with common open source assessment tools installed and configured.

Agenda - 2 days

Basic architecture of iOS, including platform security features and services

Setting up an assessment environment, including network interception and device instrumentation

Best practices for storage of data on mobile devices, how to determine whether these are being followed, and the potential consequences of not doing so

Secure network communications and common pitfalls with HTTPS web services

Run-time instrumentation and manipulation of applications

Anti-tampering controls, how they work, and how to bypass them

An example application assessment, providing an opportunity to put these methods into practice in a free-form, unguided environment

Security in Software Development Lifecycle

Do you develop software? Whether bespoke or off the shelf, does it put your customers at risk of a security breach? Do you choose components for your enterprise architecture with security in mind and an awareness of how they impact your exposure? Beyond a pre-release penetration test, do you follow secure development best practices throughout the product's lifecycle?

This course demonstrates the approach you should take in planning and developing a secure software development lifecycle.

Who should take this course?

This course is aimed at senior software developers, QA engineers, software architects, technical project/ product/program managers, business analysts and team leaders who want to understand how to satisfy expectations around security and privacy for software and hardware over which they have responsibility or liability.

Requirements

There is no requirement to have programming skills, however, a technical understanding would be beneficial to follow secure design principles and architecture decomposition.

One computer per delegate running Windows that you have the rights to install applications on.

Deliverables

Modular PDF slide-deck of the material covered during the training

Answers to the threat modelling exercises

Agenda - 3 days

SDLC place in organisation's security program.

Maturity models.

Types of SDLC: waterfall, agile, lean etc.

Stages of SDLC: requirements gathering, architecture and design, development, testing/ validation, release/maintenance.

Detailed coverage of security activities suitable for each stage.

Software-centric threat modelling.

Analysing and decomposing the application.

Applying STRIDE to identify potential threats.

DREAD and other methods of prioritisation.

Determining the countermeasures and mitigations.

Secure Coding in Java for Web Developers

NCC Group assists enterprises and product companies in improving the security of the software they develop. We offer these secure coding training courses to help developers promote a security mind-set and eliminate security flaws before they are created

NCC Group offers the most effective secure coding training available based on years of experience reviewing client source code and improving the security of the software they develop. The instructor-led Secure Coding in Java for Web Developers course provides developers with practical guidance for developing secure web applications using Java technologies.

Who should take this course?

This course combines information from the OWASP Top Ten Most Critical Web Application Security Risks with secure coding rules from the CERT® Oracle® Secure Coding Standard for Java. The course consists of lecture, demonstrations, and labs all using examples from the Java programming language and associated Java technologies.

Participants should come away from the course with a working knowledge of how to secure code web applications using Java technologies and avoid, prevent, and mitigate against the top ten most critical web application security risks

Moreover, the course encourages programmers to adopt security best practices and develop a security mind-set that can help protect software from tomorrow's attacks, not just todays.

Requirements

The course assumes basic web and Java development skills but does not assume an in-depth knowledge of software security. Course demos and solutions to exercises are presented using the IntelliJ IDE, but students are free to use any IDE for reviewing example code and performing exercises.

Participants must bring a properly configured laptop computer equipped with the following:

- Java SE Development Kit 8
- Java SE Development Kit 11
- Java 11 compatible IDE
- 100MB or greater of free hard disk space
- Current version of Adobe Reader

Participants will also receive course slides, exercises, demos, and reference materials.

Deliverables

The CERT Oracle Secure Coding Standard for Java and Java Coding Guidelines: 75 Recommendations for Reliable and Secure Programs books authored by Robert C. Seacord and published by Addison-Wesley will be provided.

Participants will also receive a DVD containing course and reference materials.

Agenda - 1 day

Explain the need for secure coding.

Follow fundamental secure coding guidelines.

Top ten most critical web application security risks and how to mitigate them in Java web applications

Injection

Broken Authentication

Sensitive Data Exposure

XML External Entities (XXE)

Broken Access Control

Security Misconfiguration

Cross-Site Scripting (XSS)

Insecure Deserialization

Using Components with Known Vulnerabilities

Insufficient Logging & Monitoring

Beyond the Beast: Deep Dives into Crypto Vulnerabilities

This training is focused on drawing out the foundations of cryptographic vulnerabilities. These topics are timeless, and when the last application using ECB or CBC mode has upgraded - they'll be the foundations of the next evolution of impactful and popular cryptographic vulnerabilities. We'll talk about attacks in the past that took advantage of cryptographic vulnerabilities. Additionally, we'll look at how algorithms and protocols have evolved over time to address these concerns, what they look like now, and where they are found in respect to the most popular bugs today. The other major areas we hit are cryptographic exploitation primitives such as chosen block boundaries, and more protocol-related topics, such as how to understand and trace authentication in complex protocols.

This training is an extension of our research done day-in and day-out in the field. This includes sharing developing proofs of concept of attacks and teaching cryptography to anyone interested.

Who should take this course?

This course is targeted at students who have a strong interest in cryptography and some measure of cryptographic understanding (such as the difference between symmetric and asymmetric crypto). The ideal student has investigated one or more recent cryptographic attacks deeply enough to be able to explain it but has not sat down and read PKCS or NIST standards describing algorithm implementation. No explicit understanding of statistics or high-level math is required, as the focus is on the underlying causes of the vulnerabilities. Some small experience of programming is recommended.

Requirements

Participants are expected to have some familiarity and efficiency in a programming language of their choosing

A laptop prepared with a programming environment they are comfortable in

Deliverables

Our trainers will provide course materials, slides, cheat-sheets and exercises of example attack implementations.

Agenda - 2 days

The right and wrong questions when discussing cryptography - why focusing on matching key lengths isn't going to find you something exploitable and what will.

Randomness, unpredictability, uniqueness. It covers the requisite info on spotting Random vs SecureRandom, but quickly dives deeper and talks about why randomness, uniqueness, and unpredictability are so important for constructions like GCM and stream ciphers, CBC and key generation).

A focus on integrity covering AEAD modes, how to use them safely and how to exploit them, disk encryption, encrypt-then-mac, and unauthenticated modes like ECB/CBC/CTR.

Complicated protocols and systems deployed at scale - how to trace through them, following how trust is granted, its scope, how it can be impersonated, and how the system falls apart when anything is slightly off.

Math and how it leads to some of the most impressive attacks. We look at several standards, many provably secure, and show how a slightest missing sanity check allows for an often-devastating adaptive chosen ciphertext attack on RSA, DSA, ECC, and unauthenticated block cipher modes.

Side channels, going in depth on the two aspects of cryptographic oracles: how the oracle is exposed and how to take advantage of what it tells you. We cover timing, error, and the CPU cache, starting off showing how to apply the attacks you've learned, and then moving on to show how to extract key bits from hand-optimized algorithm implementations.

Processor, OS & Compiler Foundations

This is a course that teaches the methodologies to understand native code from both a static analysis perspective using disassemblers and a dynamic perspective using debuggers. It also covers some common classes of security vulnerability and methods to detect them.

The course is run over two days but the second day covering the black-box assessment can be omitted when the course is used as an introduction to the Exploit Development course.

Who should take this course?

This course is suitable for consultants, code reviewers, reverse engineers and exploit developers who want to understand how native programs work and assess their security without access to source code.

Requirements

All the examples on the course are 32-bit Windows executables. Participants are expected to have some familiarity with the C language and be capable of writing and compiling simple programs. Knowledge of the Win32 API is an advantage but not a requirement

Participants require a laptop running Windows 7 or above with local administrator rights.

The following free packages are required:

- Visual Studio 2015 Community Edition
- Debugging Tools for Windows
- OllyDbg or Immunity Debugger
- IDA (freeware edition)

Deliverables

PDF presentation of over 150 slides split into the nine modules covered during the training.

Zip files of practical exercise, model solutions and reference material.

Agenda - 2 days

Introduction to x86 assembly language

Compiling a simple program, disassembling it and running it under a debugger.

Introduction to debugging techniques and the various tools available.

Using a debugger to solve a series of "crackme"-style challenges.

Introduction to reverse engineering using IDA.

Using a disassembler to understand a series of "reverseme"-style challenges.

Understanding common coding errors in C.

Combining the tools and techniques to perform a black-box vulnerability assessment.

Exploit Development

The is a course that teaches various exploitation methods from simple stack overflows to type confusion bugs in C++ code using a variety of techniques including return oriented programming and engineering read/write primitives.

The instructors for the course come from the NCC Group Exploit Development Group (EDG) which provides bespoke exploits and tools for use on client engagements. They have been working in vulnerability research for over 15 years in a variety of roles.

The EDG has developed exploits against popular software including Internet Explorer, Firefox, Flash, Adobe Reader, Windows Kernel, Xen and Java. Their exploit development skills are backed up with extensive knowledge and experience of reverse engineering and low-level debugging.

The course is run over three days but the second day covering the payload development can be omitted if desired.

Who should take this course?

This course is aimed at consultants, code reviewers, reverse engineers and exploit developers who want to understand how vulnerabilities in native code can be exploited.

Requirements

Participants are expected to have some familiarity with x86 assembly language and be comfortable with assembly level debugging. Experience with a scripting language such as Python, Perl or Ruby is highly recommended.

Participants require a laptop with local admin access running a recent version of VMWare.

The following free packages are required:

- A debugger: Debugging Tools for Windows, OllyDbg or Immunity Debugger.
- A disassembler: IDA (freeware edition) strongly recommended
- An assembler: NASM or FASM.
- A scripting language: Python, Perl or Ruby.

Deliverables

PDF presentation of over 250 slides split into the 13 modules covered during the training.

Zip files of practical exercise, model solutions, reference material and all necessary tools.

Agenda - 3 days

Stack overflows

Writing a simple exploit

Mitigations

Return oriented programming

Developing a ROP exploit

Developing payloads and shellcode

Common filters and writing filtered exploits

C++ internals

Exploiting vtable overwrites

Understanding type confusion (casting bugs and use-after-free), and in-depth exploitation of a casting bug

Windows Kernel Exploitation

This course demonstrates the approach you should take in attacking a previously unknown component in the Windows kernel. After detailing the Windows kernel internals applicable to exploiting such a vulnerability, the focus is on labs that teach you what it takes to exploit a real-world vulnerability.

This class focuses on exploiting CVE-2018-8611 on Windows 10 x64 1809 (RS5), a fairly complex race condition that leads to a use-after-free on the non-paged kernel pool. The vulnerability is in the Kernel Transaction Manager (KTM) driver (tm.sys), a kernel component that has not yet received much public scrutiny.

Who should take this course?

NCC Group's Windows Kernel Exploitation training is a focused course for reverse engineers, exploit developers and bug hunters who wish to learn a structured and reusable approach to attacking an unknown component in the Windows kernel.

The lab exercises utilise the KTM component extensively, however the focus of the course is on learning an adaptable methodology that can be reused on further components you encounter in the future. The emphasis of the course is the thought process behind exploring functionality, identifying unmitigated code paths and abusing bugs.

Requirements

Participants are expected to have some familiarity with the C programming language, x86/x64 assembly

Familiarity with common memory corruption techniques and userland exploitation on Windows or Linux

Comfortable with both disassemblers (IDA, Ghidra, etc.) and debuggers (WinDbg, x64dbg, gbd, etc.)

Recommended setups are: Linux, MacOS or Windows base OS with VMware, at least 80GB of free disk space and 8GB RAM

Deliverables

Modular PDF slide-deck of the material covered during the training

Two VMs are provided – a vulnerable VM and a debugger/development VM which is optional if your base OS is Windows

Agenda - 3 days

Windows kernel fundamentals

Binary diffing Microsoft updates

Kernel Transaction Manager (KTM) basics

Understanding CVE-2018-8611

Labs – debug environment setup, binary diffing, KTM experimentation and triggering the vulnerability

Exploitation techniques

Labs – triggering in a debugger, bad vs good feng shui, race win detection, exploitation strategies, discovering a kernel leak and restoring cleaned execution

How to escalate privileges

Labs – Arbitrary read and write primitives, privilege escalation, alternative read and write primitives